CERN

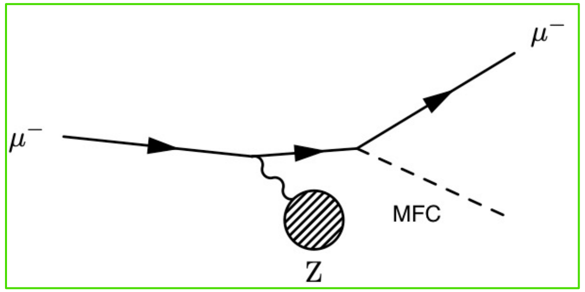
Generating muonic force carriers events with classical and quantum neural networks

Michele Grossi, Enrique Kajomovitz, **Oriel Kiss**,
Tigran Ramazyan and Sofia Vallecorsa

Motivation and Overview:

- Use generative models to **sample efficiently** from the underlying distribution of a data set.
- Enables us to sample directly as **input of the conditions** (not like Mad-graph where it has to be fixed from the start).
- The use case is the generation of muonic force carriers (MFCs) events, trained on Mad-Graph simulations.
- We consider classical Conditional GANs and quantum Born machines for this task.

Muonic Force Carriers

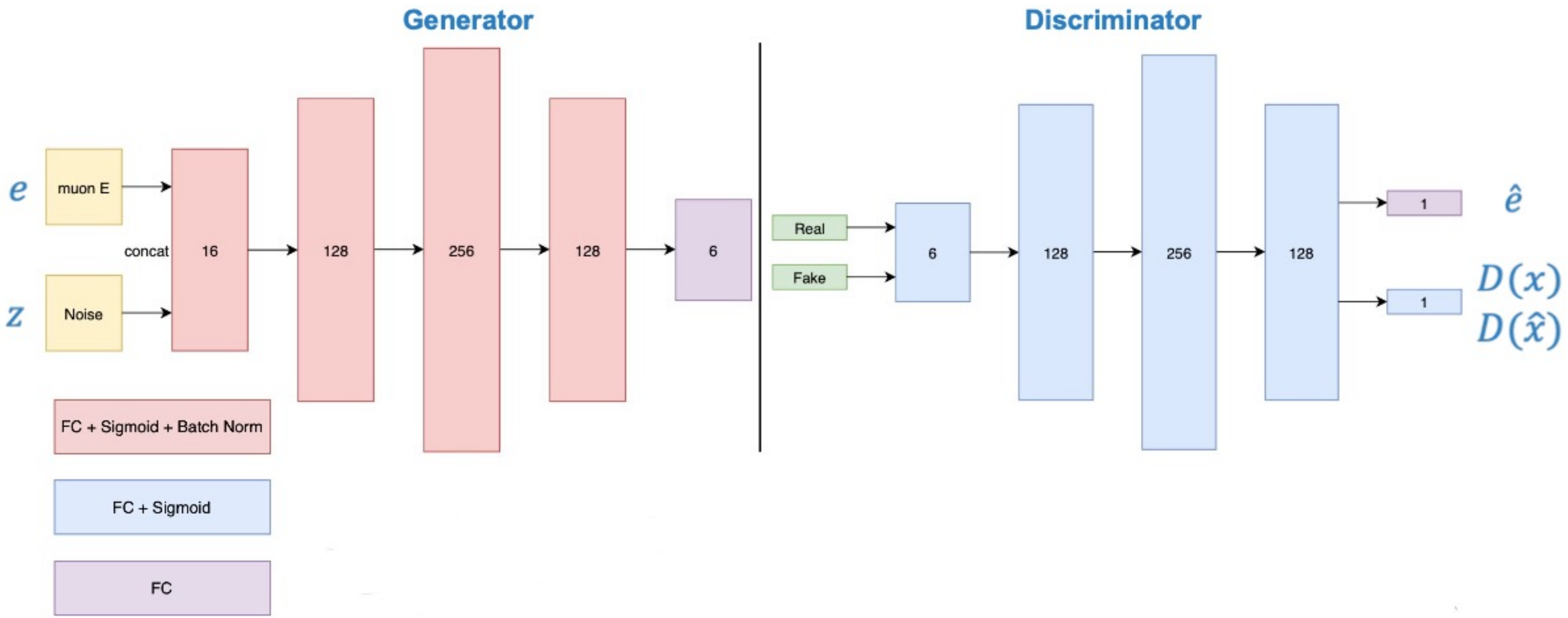


- MFCs appear in different theoretical hypothesis, as a constituent of **dark matter** and could explain the **anomalous magnetic dipole** moment of the muon or the anomaly in the measurement of the **proton radius.**
- 2 use cases: a) muon fixed-target collision (**FASER**).
  b) muon interactions in the **ATLAS**[1] calorimeter.
- We are interested to generates following features for the outgoing muon and MFC: **energy** (E), **transversal momentum** (pt) and **pseudorapidity** (η).

[1] Galon, I, Kajamovitz, E et al. "Searching for muonic forces with the ATLAS detector". In: *Phys. Rev. D 101, 011701* (2020)

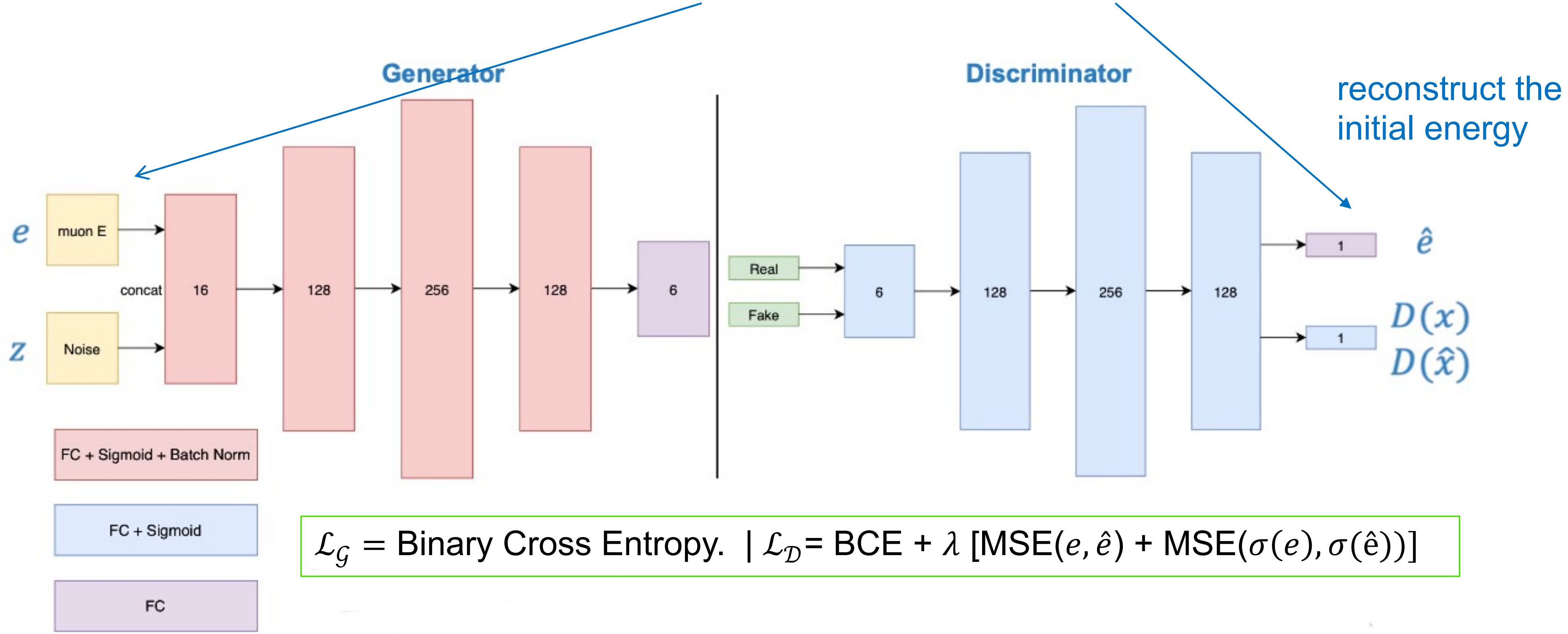
Conditional Generative Adversarial Network (C-GAN)

The topology has been optimized for this task.



C-GAN: where is the condition?

The data set is binned in function of the incoming muon's energy



$\mathcal{L}_{\mathcal{G}}$ = Binary Cross Entropy. | $\mathcal{L}_{\mathcal{D}}$= BCE + λ [MSE($e, \hat{e}$) + MSE($\sigma(e), \sigma(\hat{e})$)]

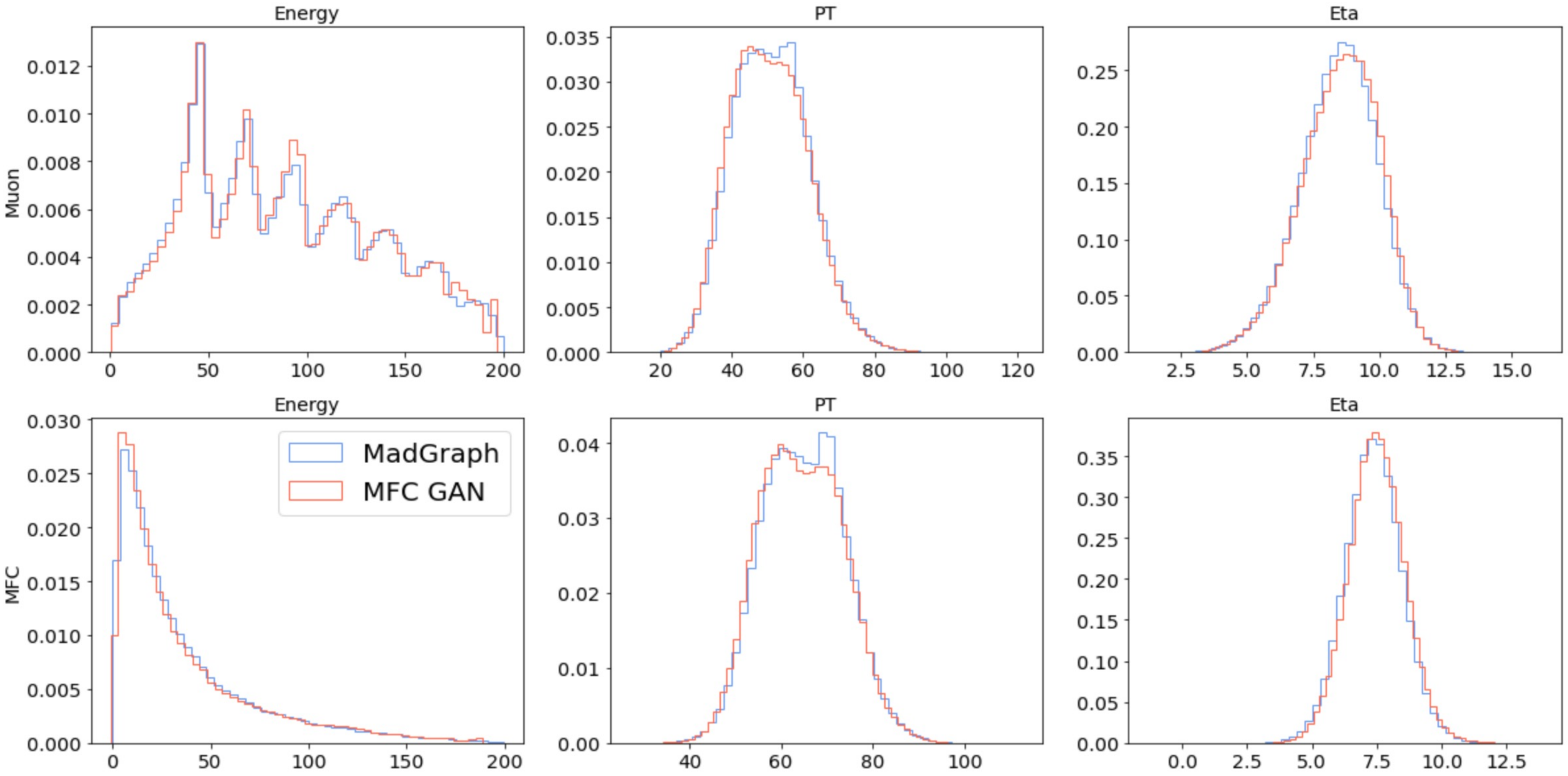
C-GAN: full results

- Preprocessing: energy scaling, power transform on pt, standard scaling.
- Hyper parameters tuning.

$$TV = \frac{1}{2I}\sum_{i,x}|P_i(x) - Q_i(x)|$$

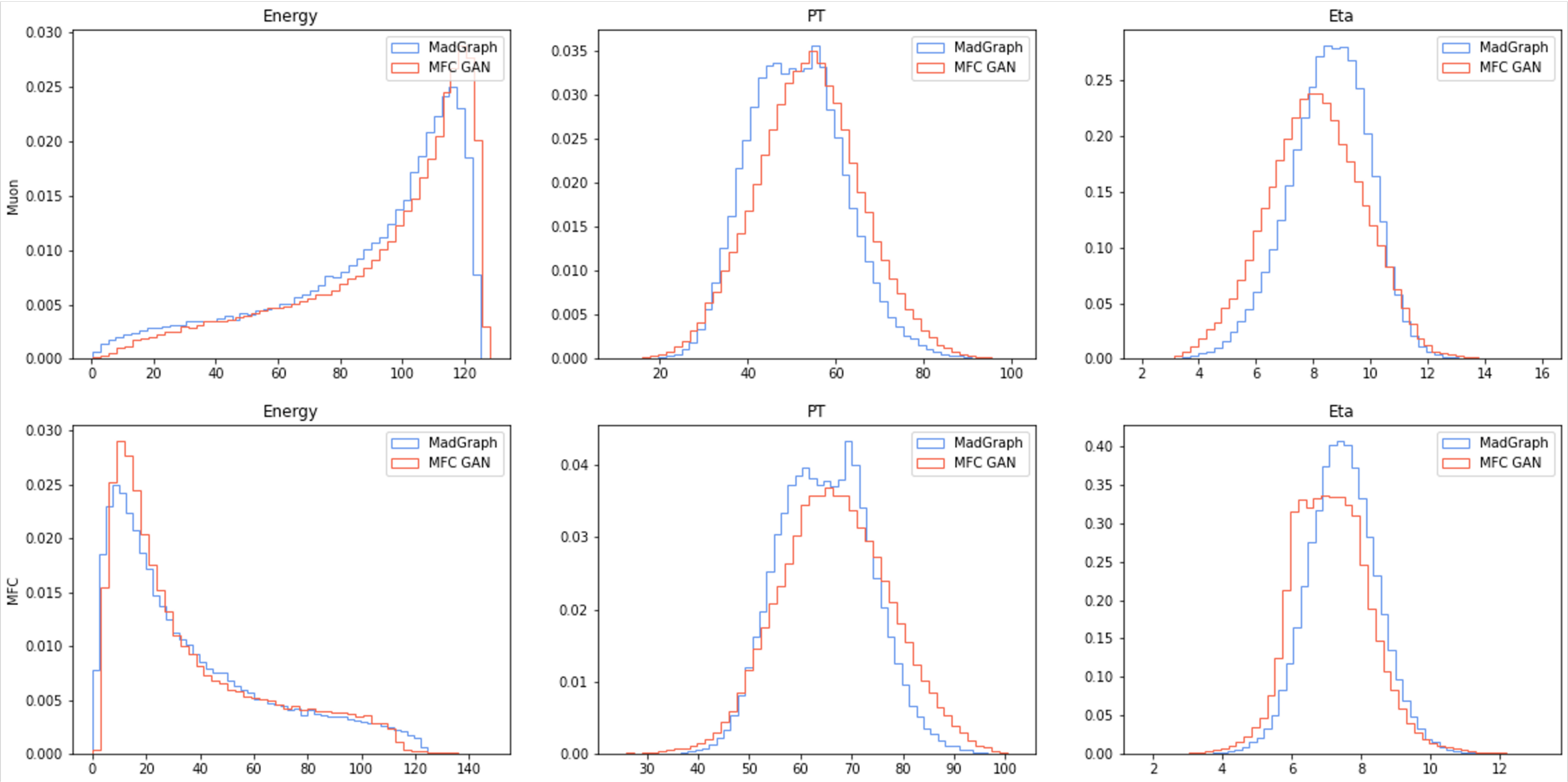
TV = 0.032

I = #features



C-GAN: Interpolation

125 GeV (not in the training set)



Variational Methods: basic concept and tools

The Variational Quantum Algorithms Blend Quantum and Classical computation in order to keep the circuit shallow making use of an **optimization based/learning based** approach.

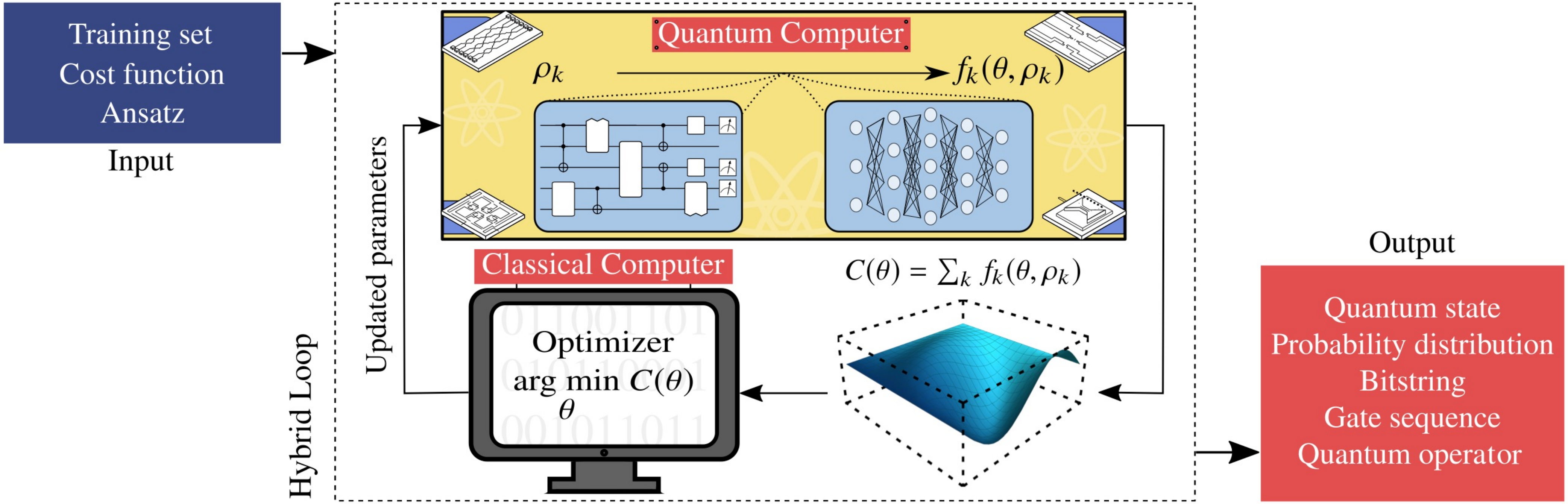
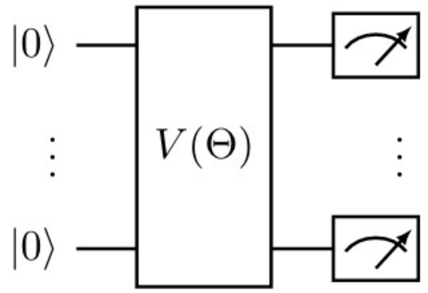


Figure 1. **Schematic diagram of a Variational Quantum Algorithm (VQA)** [2]

[2] Variational quantum algorithms, *https://arxiv.org/pdf/2012.09265.pdf*

Quantum Born machine[3]

- *Sample from a variational wavefunction $|\psi(\theta)\rangle$ with probability given by the Born rule: $p_\theta(x) = |\langle x|\psi(\theta)\rangle|^2$*



n dimensional binary strings map to 2^n bins of the discretized dataset

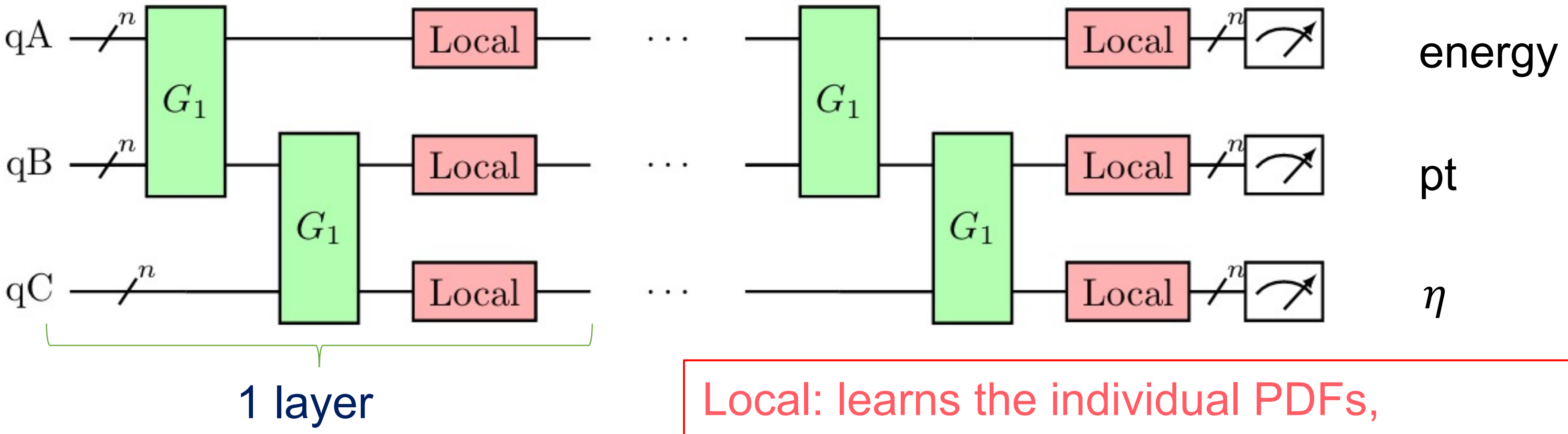
- *Only able to generate discrete PDFs (continuous in the limit #qubits $\to \infty$)*
- *Maximum Mean Discrepancy: MMD(P,Q) = $\mathbb{E}_{\substack{X\sim P\\Y\sim P}}[K(X,Y)] + \mathbb{E}_{\substack{X\sim Q\\Y\sim Q}}[K(X,Y)] - 2\mathbb{E}_{\substack{X\sim P\\Y\sim Q}}[K(X,Y)]$ with K a gaussian kernel $K(x,y) = e^{-(x-y)^2/\sigma}$ with $\sigma \in [0.1,1,10,100]$*

- *Pros: low sample complexity, Cons: weak convergence properties*

[3] Coyle, B., Mills, D. et al, "The Born supremacy". In: *npj Quantum Inf 6, 60* (2020)

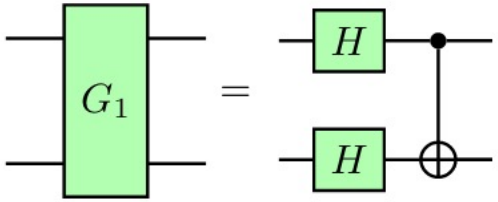
Born machine: multiples features[3]

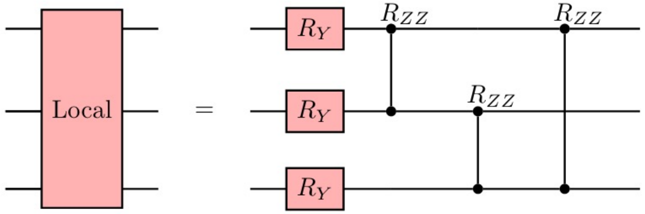
Use multiples quantum registers



Local: learns the individual PDFs,
time evolution of an Ising type Hamiltonian,
conjectured to be difficult to simulate classicaly.

G_1: creates a Bell state (maximally entangled state) between the first qubit in each register.





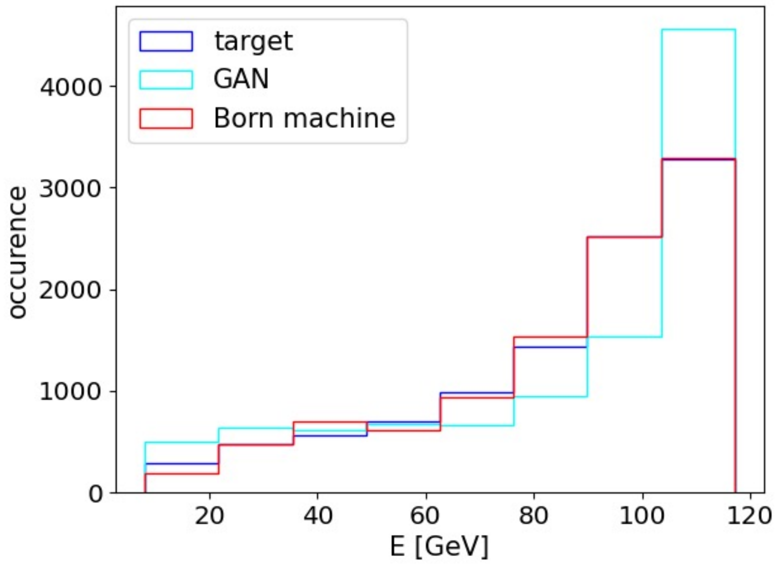
[4] Elton Yechao, Sonika Johri et al, "Generative Quantum Learning of Joint Probability Distribution Functions" In: arXiv 2109.06315

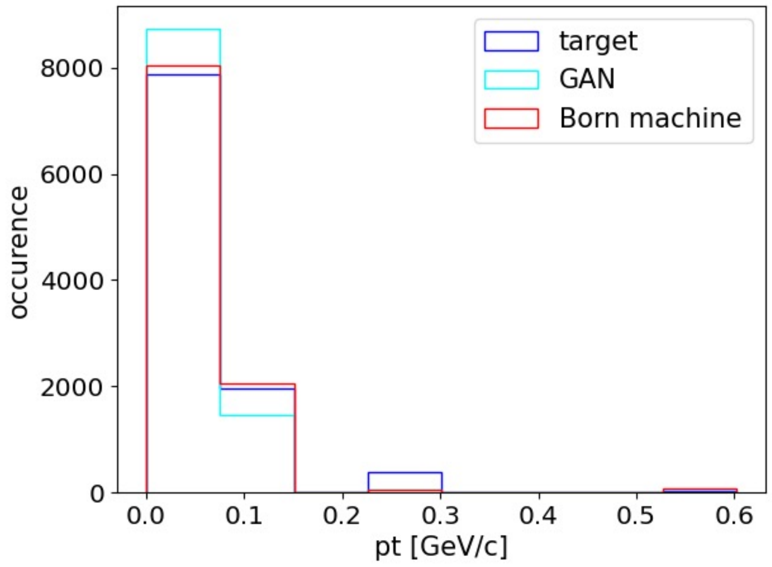
Results

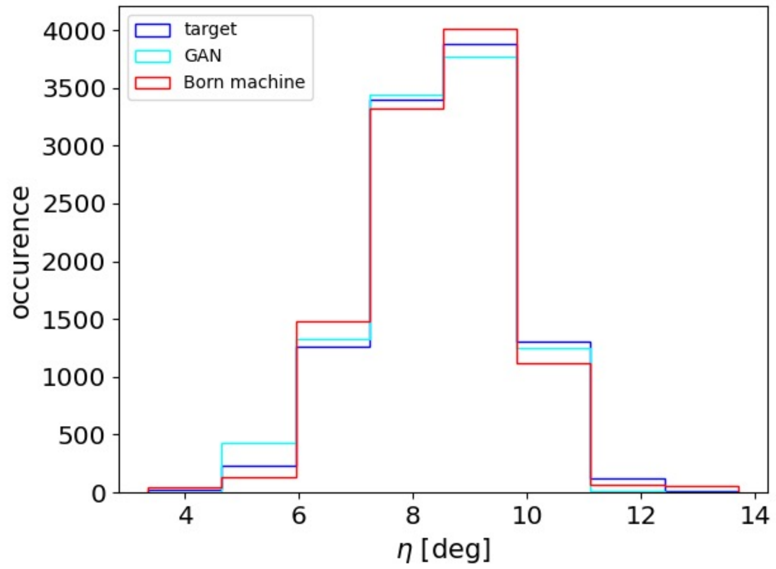
Born and CGAN (as a regression task) retrained on the ***same discretized dataset****.*

Outgoing muon with initial energy of 125 GeV.

Preprocessing: min max scaling.







Total variance: TV(P,Q) = $1/2I \sum_{i,x} |P_i(x) - Q_i(x)|$.

I = #features

$TV_{GAN} = 0.09 \quad TV_{BORN} = 0.03$

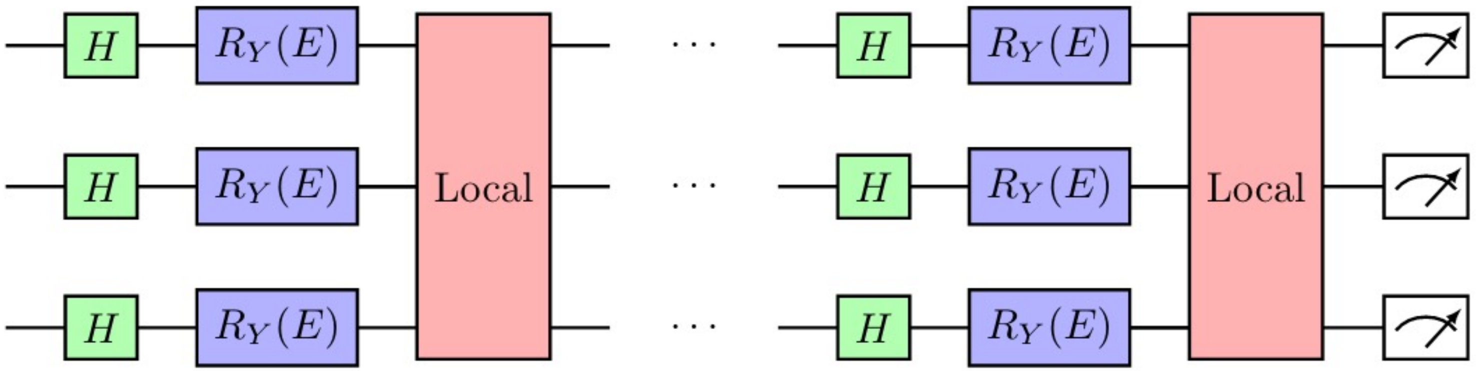
Conditional Born machine

green: fixed gates
blue: data encoding gates
red: trainable gates

Data encoding: via data-parametrized rotations

Input: binning energy E (scaled between $[-\pi, \pi]$)

Interpolation: train only on certain energy bins and the model should learn to predict in between.



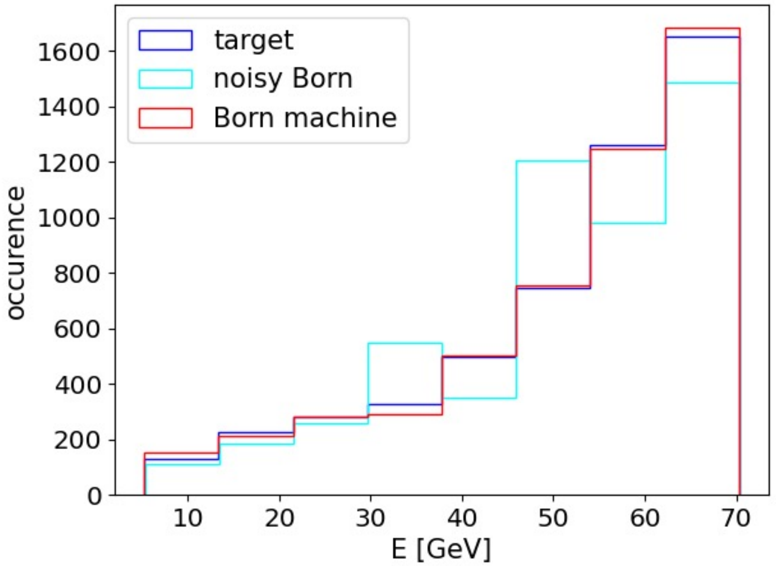
Data re-uploading makes the quantum circuit more expressive as function of the data.

C-Born machine: results

Outgoing muon energy

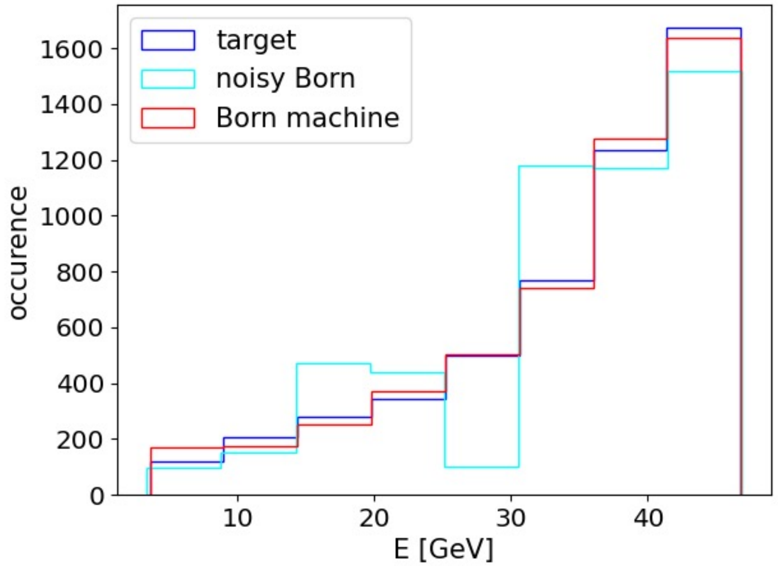
Noise: mimic exact hardware (gates errors, readout errors). Taken from IBMQ casablanca

a) train: 150 GeV



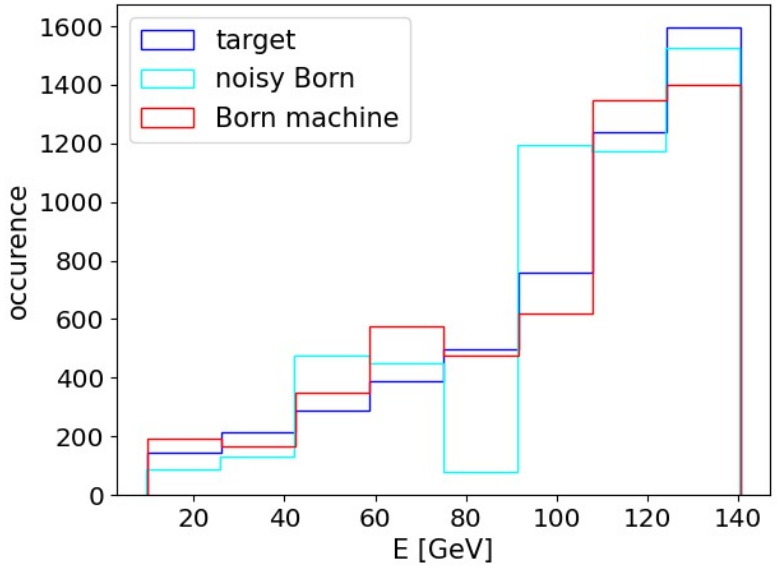
TV = 0.03, TV_noise = 0.12

b) train 200 GeV



TV = 0.02, TV_noise = 0.15

c) test 175 GeV



TV = 0.07, TV_noise = 0.14

Conclusion:

- Use C-GAN and Born machine to generate MFC events.
- C-GAN is able to generate all features, interpolation is ongoing work
- The Born machine is currently able to learn discrete distributions because of the hardware limitations.
- The Born machine is able to interpolate.
- Future works would be devoted to real hardware and larger physical systems

Thanks for your attention!

oriel.kiss@cern.ch
sofia.vallecorsa@cern.ch

CERN | QUANTUM TECHNOLOGY INITIATIVE

Backup: CGAN training

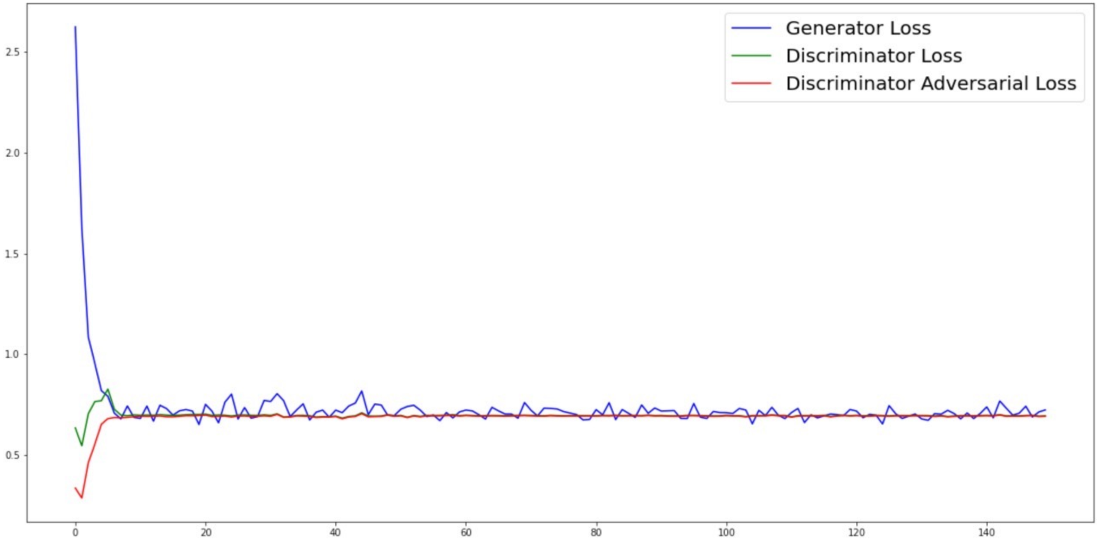
Data Preprocessing:

- E = E/E_ingoing
- Pt = $pt^{0.2}$
- Standard scaling

Hyper-parameters tuning:

- learning rate
- number of epochs
- network's topology
- auxiliary loss weight

Training loss with RMSProp



Wasserstein CGAN did not work well

CERN